STANDARD SPACECRAFT INTERFACES AND IP NETWORK ARCHITECTURES: PROTOTYPING ACTIVITIES AT THE GSFC

Richard Schnurr, Jane Marquart, Michael Lin
NASA Goddard Spaceflight Center

ABSTRACT (100 words)

Advancements in flight semiconductor technology have opened the door for IP-based networking in spacecraft architectures. The GSFC believes the same significant cost savings gained using MIL-STD-1553/1773 as a standard low rate interface for spacecraft busses can be realized for high-speed network interfaces. To that end, GSFC is developing hardware and software to support a seamless, space mission IP network based on Ethernet and MIL-STD-1553. The Ethernet network shall connect all flight computers and communications systems using interface standards defined by the CCSDS Standard Onboard InterFace (SOIF) Panel. This paper shall discuss the prototyping effort underway at GSFC and expected results.

INTRODUCTION

This paper describes the GSFC activities to develop prototype software and hardware for a Flight Ethernet-based Spacecraft implementing an onboard IP network. The network is patterned after a network stack architecture defined by the CCSDS Standard Onboard InterFaces (SOIF) activity. The goal of SOIF is to develop and promote standard onboard networks and enable the development of interoperable space hardware components. The GSFC prototype activity will develop and demonstrate a seamless space mission network based on UDP/IP, Ethernet and MIL-STD-1553.

The prototype supports missions requiring high-speed data busses with ground-based interface standards. The first phase of the prototype activity is a proof-of-concept for the Global Positioning Measurement (GPM) mission, which has base-lined Ethernet for its spacecraft bus. This phase will include the modification of heritage flight software to support IP onboard networking and communications with the ground. It will also use commercial Ethernet hardware to demonstrate a fully functional C&DH system. The second phase will replace four of the commercial Ethernet interfaces and the Ethernet switches with flight prototypes developed by the ESTO/SC&DS SpaceLan technology activity. It is expected that the prototype will quantify the reductions in development and test costs possible by using commercially compatible Ethernet interface standards in an onboard IP network architecture. It will also demonstrate enhanced flexibility, portability, and software and hardware reuse among missions.

Following the initial prototype, the onboard network activity will continue development of a more comprehensive onboard network that addresses issues such as managing time-critical data (isochrony) and quality of service components. In addition, a MIL-STD-1553 bus and related software will be added to the prototype.

ARCHITECTURE

Overview: The onboard network architecture is directly drawn from the CCSDS SOIF implementation model recommended for space mission data systems. The model terminology is consistent with the OSI Basic Reference Model and includes the Physical, Data Link, Network, Transport, and Applications Layers, shown in figure 1 (reference 1).
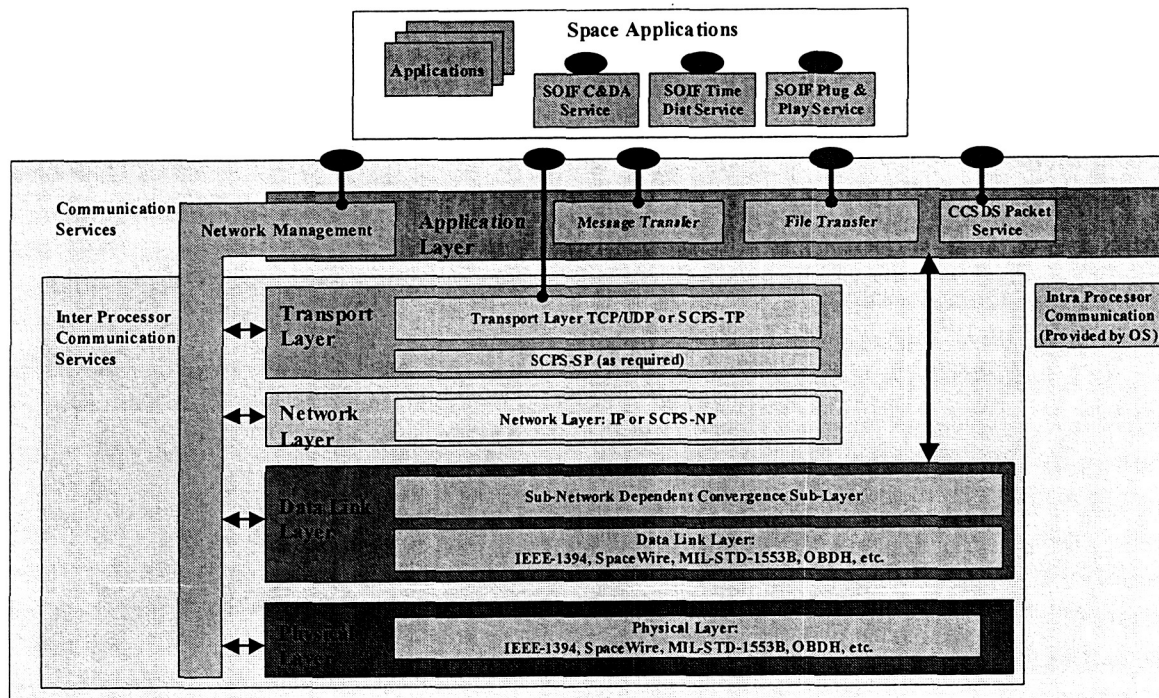

Figure 1. SOIF Implementation Model

Many of today's spacecraft follow CCSDS protocols, specifically, routing data as CCSDS telemetry packets and CCSDS telecommands. When using MIL-STD-1553 as the Data Link, a CCSDS data packet generated by the application layer is directly routed to the data link layer, following the path of the vertical black arrow in figure 1.   No standard transport or network layer is used on the MIL-STD-1553 and Spacelink.   SOIF advocates using a standard network and transport to provide connectivity for sub-networks using different Data Links. To achieve interoperability, standard Convergence sub-layers are required for each Data Link. One of the long-term goals of SOIF and the GSFC prototype activity is to develop and coordinate convergence sub-layers for Ethernet and MIL-STD-1553. A convergence sub-layer for Ethernet will be developed and demonstrated in the testbed.

The GSFC has baselined IPv4 as the network layer protocol for both the testbed and the GPM spacecraft bus. The GPM spacecraft bus architecture includes two command and data handling systems (C&DH), two communications systems, and redundant power and ACS subsystems. In addition, GPM supports two MIL-STD-1553 busses, one for instruments and one for ACS

components. The prototype IP network, implemented within the context of GPM, is shown in Figure 2.
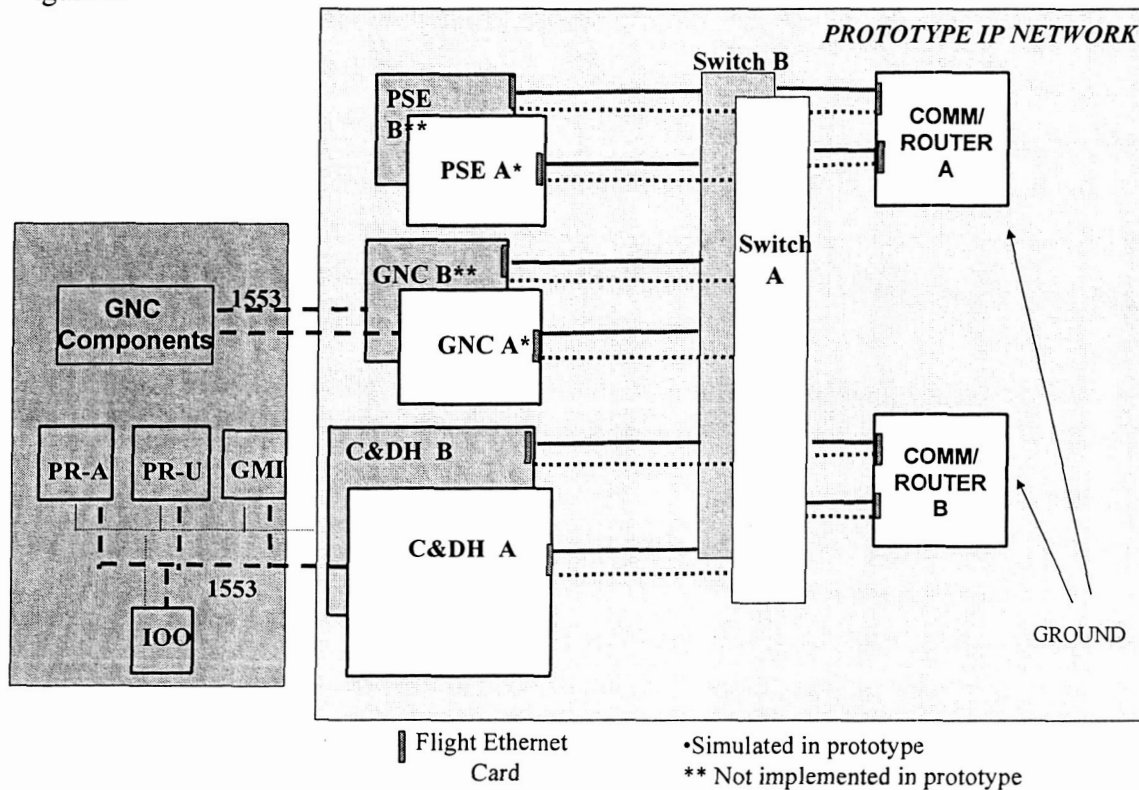

Figure 2. Prototype Spacecraft Bus

The prototype C&DH software is based on the New Millenium Program ST-5 C&DH architecture that uses an inter-task messaging system called the Software Bus. A network bus service, developed for the prototype, connects the Software Bus to the network for inter-processor messaging. When architecting the network used for spacecraft communications, two transport options are available, TCP and UDP. Neither provides a comprehensive solution to the onboard network requirements. While TCP provides reliability, the timeliness of the reliability is not sufficient for mission critical, real-time applications. UDP is similar to the transport methods used today, however, characteristics such as packet-retry, and in-order delivery are lacking. Solving the issues associated with UDP is straightforward; hence it was selected as the transport. Data is routed as UDP/IP packets onboard the spacecraft and the enhancements necessary to meet flight requirements are added at the Application Layer. This allows a standard IP stack to be used in the prototype. The device drivers, developed for the prototype, interface the network stack to both commercial and flight Ethernet cards. The flight Ethernet card uses a standard MAC core with a custom LVDS physical layer interface, enabling the use of commercial network components and software for development, test, and ground operations, the core premise for cost savings. Both the hardware and software components are detailed in the following paragraphs.

Hardware: Ethernet is one of the most common computer network Physical and Data Link layers in use today. It has evolved over the years from a Collision detect, Multiple access, half duplex, 10 Mbps interface to a high speed network using cross point switches and full duplex twisted pair physical interfaces at data rates up to 1 Gbit per second. It is this switched full duplex topology that

enables Ethernet to meet Spacecraft interface requirements such as latency while maintaining compatibility with ground networks. The current effort is restricted to 10/100 Mbit Ethernet over twisted pair and Flight Ethernet using a 12.5/125 Mbit DS Link encoded physical layer.

GSFC has developed a prototype flight PCI network card that supports connections to two Ethernet networks simultaneously. A twelve-port flight switch prototype has also been developed. The designs are to be ported to Flight FPGAs to support the GPM mission. Both the NIC and Switch support a GSFC D/S Physical interface using Low Voltage Differential Signaling (LVDS) drivers. A media converter has been developed to translate between the LVDS physical layer and Standard Ethernet. It is expected that we can support both 10 Mbps and 100 Mbps networks with the flight FPGA implementation. See figures 3 and 4 for the configuration of the flight interface card and switch respectively.
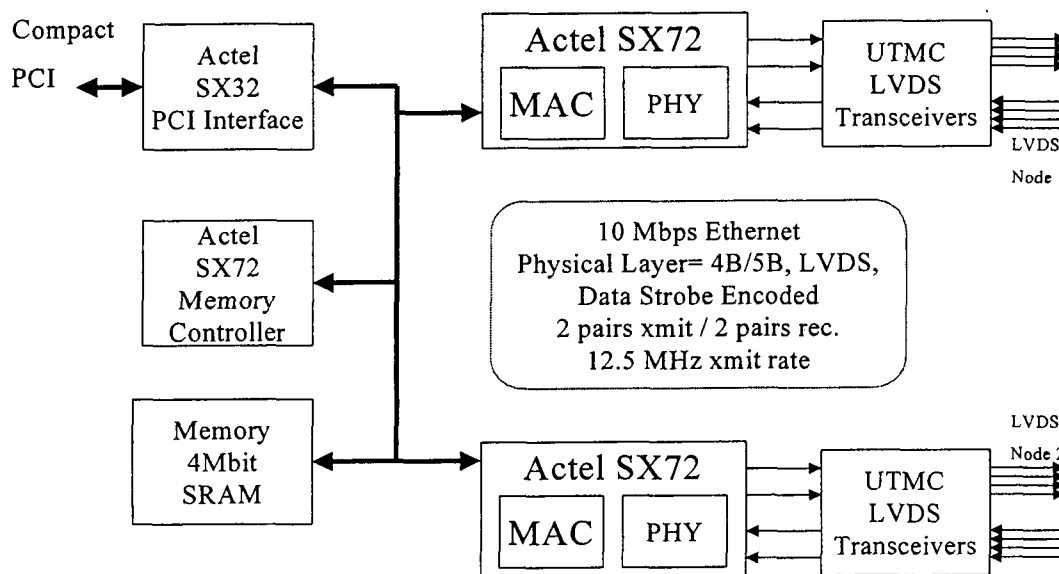

Figure 3. Prototype Flight Ethernet Card

GSFC concluded early on that using one of the existing Ethernet physical layers for flight would be a costly and time-consuming process since the development of a flight-qualified Ethernet Physical layer integrated circuit would be required. Consequently, GSFC has developed and demonstrated a Low Voltage Differential Switching Physical interface utilizing DS link encoding that supports Ethernet data transition up to 100 Mbps. The actual signaling rate on the bus is 12.5/125 Mbps with the Ethernet encoding leading to a 6.25/62.5 Mhz frequency on the D and S signals. This custom physical layer eliminates the need to develop a flight qualified Ethernet physical layer chip while allowing the use of standard Ethernet protocols and MAC cores by preserving the standard MII interface between 10/100 MAC cores and the physical layer circuits. Indeed, preserving the MII enabled the construction of our low cost media converters. The media converter is used to interface any flight system or switch port to a 10/100 Base-TX commercial network. Figure 5 illustrates the different physical layers.
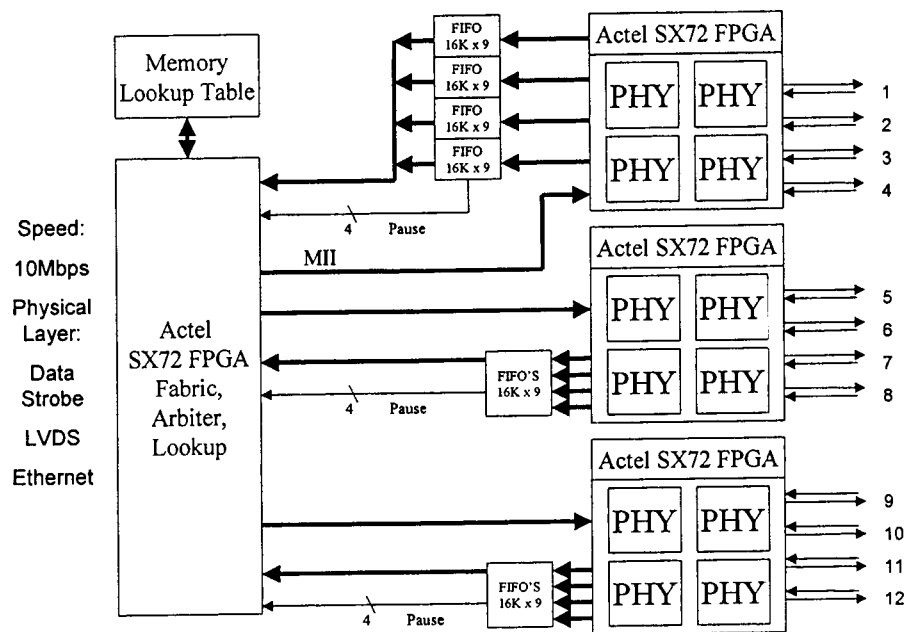
4

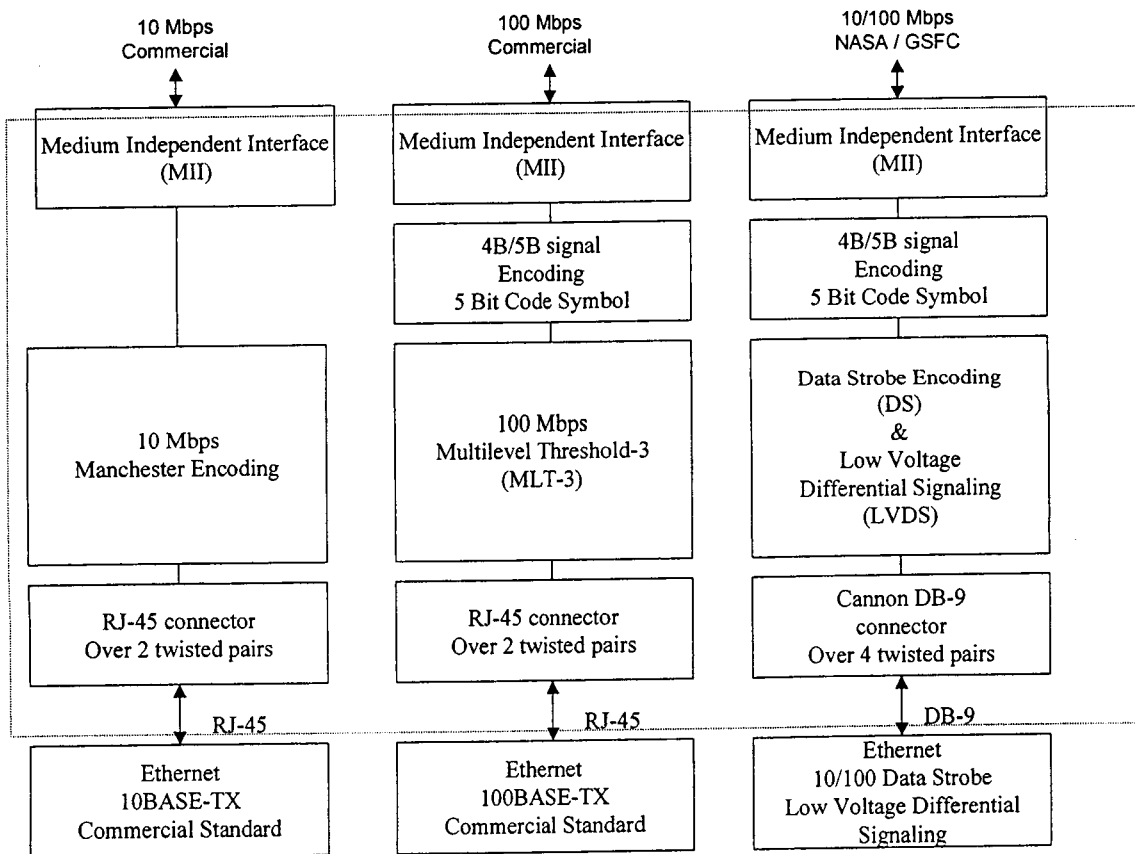Figure 4. Prototype Flight 12-Port Switch

Figure 5. Physical Ethernet Layers

The topology of the GPM spacecraft bus and our technology testbed has two independent networks each with a single string 12 port switch mirroring the testbed. Each spacecraft box is attached to both networks. Four twisted pair cables connect each network port to a switch. Thus eight twisted pairs are required to connect each flight box to both switches. A 9-pin connector is needed to support this interface.

Redundancy is very important to space missions. GSFC studies indicate that two independent networks are required to meet fault tolerance and redundancy requirements. The flight network card has two protocol engines, two physical layers, and a single PCI bus interface. Each of the redundant networks has a fully independent, single string switch. Wide latitude exists in the design of Ethernet switches. The GSFC switch supports features like fixed MAC table entries for critical systems, pause command frame processing, broadcast mode to support configuration, and support for a fixed latency mode for distributed time ticks. Spanning tree algorithms will be eliminated because of the simplicity of the onboard network topology, broadcast traffic will be sent to all ports with the exception of the port the data entered. Ethernet broadcast messages are used for time distribution and commands that must traverse a switch that has not been configured.

Software: The New Millenium ST-5 flight software, representative of typical flight architectures onboard spacecraft today, is being used as the baseline for the prototype activity. The baseline provides a real-time flight environment where performance and implementation issues of an onboard network can accurately be assessed. All application tasks, such as the command and telemetry handling, health and safety, and data storage tasks, communicate via the Software Bus service using the standard CCSDS packet format. The Software Bus also provides an interface to the open source Real-Time Operating system called RTEMS, which is responsible for task scheduling, interrupt handling, and other real-time system services. The baseline software does not include any external communications service. However, the prototype required development of an external communications service compatible with standard network and communication services. This service is called the Network Bus.

Applying the SOIF implementation model to the prototype architecture required development of a Network Bus Service. As mentioned earlier, the Software Bus (SB) service passes CCSDS packets between application tasks, based upon the App Id in the packet header. If the sending and receiving tasks reside on the same processor, the packet passes from application to application, never requiring the services of any layer below the Application Layer. If the sender and receiver are on different processors, a transport mechanism is required for transmitting the data across the bus to the receiving task, as is the case with a power system application sending its telemetry to the C&DH telemetry handling task (TO). For systems using a MIL-STD-1553 data bus, custom interface software is needed between the Software Bus service and the Data Link. Conversely, the IP/Ethernet stack only requires a Network Bus Service, described below, to interface the Software Bus to the commercially available IP stack. UDP/IP and Ethernet are supported by all modern Operating Systems, reducing the amount of custom code needed for inter-processor messaging and enabling greater portability of applications between missions with different physical busses. A long-term goal of SOIF and the GSFC prototype activity is to coordinate standard convergence sub-layers for Ethernet and MIL-STD-1553 to achieve interoperability. A comparison of the two communication stacks is illustrated in figure 6.
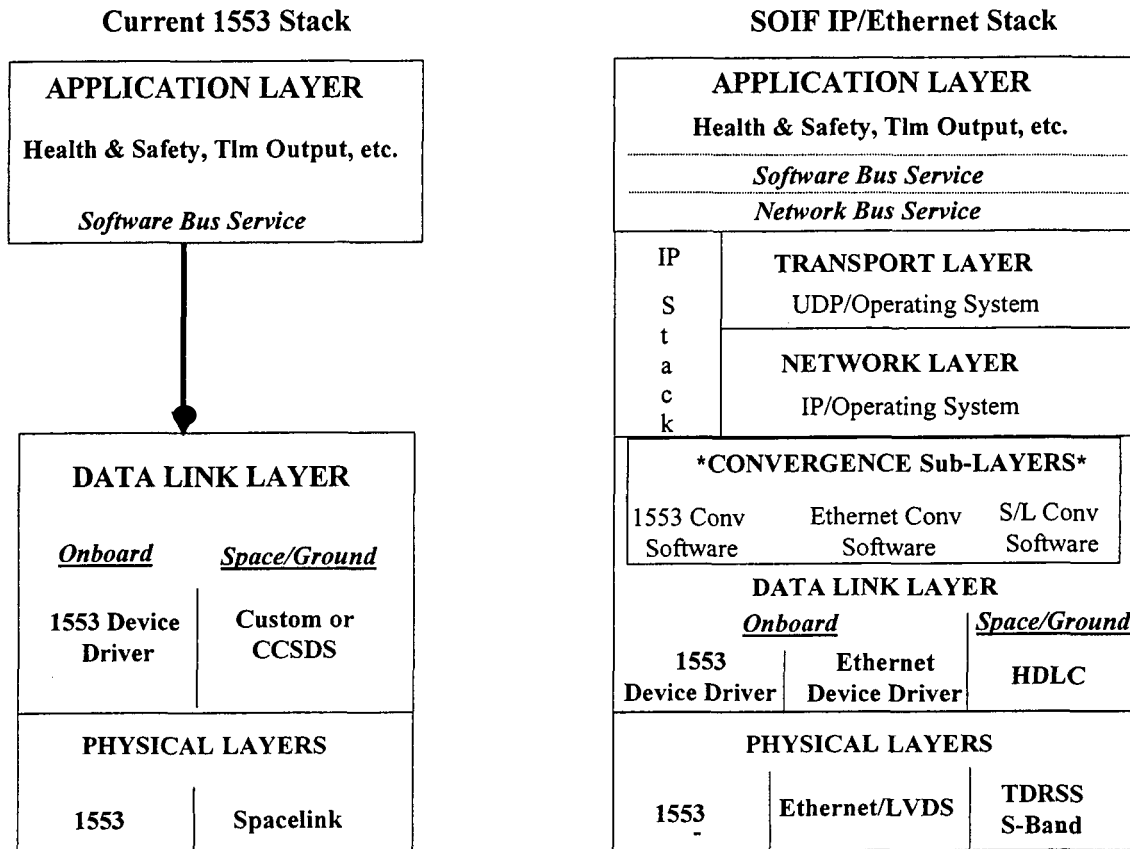
6

Current 1553 Stack   **SOIF IP/Ethernet Stack**

APPLICATION LAYER

Health & Safety, Tlm Output, etc.

Software Bus Service

APPLICATION LAYER

Health & Safety, Tlm Output, etc.

Software Bus Service

Network Bus Service

IP	**TRANSPORT LAYER**
S	UDP/Operating System
t	
a	**NETWORK LAYER**
c	IP/Operating System
k	

CONVERGENCE Sub-LAYERS

| 1553 Conv Software | Ethernet Conv Software | S/L Conv Software |

DATA LINK LAYER

| *Onboard* | | *Space/Ground* |
| **1553 Device Driver** | **Ethernet Device Driver** | **HDLC** |

PHYSICAL LAYERS

| **1553** | **Ethernet/LVDS** | **TDRSS S-Band** |

DATA LINK LAYER

| *Onboard* | *Space/Ground* |
| **1553 Device Driver** | **Custom or CCSDS** |

PHYSICAL LAYERS

| **1553** | **Spacelink** |

Figure 6. Comparison of Stacks

Network Bus Service: The Network Bus (NB) software provides peer-to-peer and client-server services needed for inter-task communication across the network. All packets destined for a different subsystem are passed from an application task via the Software Bus to NB and all incoming packets are passed from NB to the application via the Software Bus. This provides a seamless interface to applications communicating between processors. NB does not schedule its outgoing IP traffic, instead relying on the convergence/data link layer infrastructure to allocate and manage bandwidth.

NB handles UDP command and telemetry packets by separating the data into categories. Each category has a separate connection/port to NB. Categories may include data to/from other subsystems, the ground system, test equipment used during development or special types of data. For the prototype, six sources of data were defined, hence, six ports: incoming UDP ground commands; incoming UDP special commands (described below); outgoing UDP telemetry; UDP packets to/from the GNC & PSE subsystems; UDP/CLCW packets; and an optional TCP connection to the ground, which has no defined function to date. This design allows ports to be easily added or removed as the prototype evolves.

Transport/Network Layers: A commercial IP stack with no modifications is used in the Transport and Network Layers. The IP Stack encapsulates the data, in this case a software bus message, in the transport protocol UDP, and the network protocol IP and vice versa.

Convergence Sub-Layer: SOIF has developed the concept of a convergence sub-layer between the top of the Data Link layer and the bottom of the network. The need for this "Shim" layer is driven by the wide variety of capabilities provided by standard network Data Link /Physical layers, the fixed Service provided by IP and SCPS-NP, and the QOS requirements typically imposed on flight networks. While Ethernet has a well-established IP interface and would not necessarily require a Convergence sub-layer for ground applications, a flight Ethernet network may require a convergence sub-layer to implement flight level QoS. For example, schedule-driven communications, time distribution, and asynchronous message priority would be implemented in this layer. Reliability could be implemented in the convergence sub-layer or the application layer. Studies are ongoing to decide which will be used on GPM. The typical spacecraft defines the QOS on a per connection basis. For example, time distribution requires a message be delivered to users, with defined latency, from the time source. Another user might request a connection using reliable service that provides a finite retry capability within a defined time interval. Switched Ethernet provides a highly capable, reusable network fabric that when properly scheduled can provide near real-time performance. While this is not an area of concentration in the prototype, it will provide a testbed for future development.

GSFC has extensively studied flight quality of service for the JWST, SDO, and GPM missions. The following service classes cover most, if not all flight needs: Asynchronous with priority (options: reliability, segmentation), and Isochronous with scheduled time slot(options: reliability(notification), finite retry, segmentation). The major challenge in implementing the shim layer is in defining a standard slot-scheduling scheme that covers all flight applications. The other challenge is the lack of support for QOS on some network Data Link/Physical Layers. Unfortunately, Ethernet and MIL-STD-1553 do not have standard QOS implementations. The GSFC efforts in this area will revise and extend our successful schedule table concept used on our MIL-STD-1553 bus implementations.

Device Driver: The device driver software was developed in two phases, first in Linux for checkout of the hardware. The driver was then ported to RTEMS, a Real-time Operating system, and customized for the flight Ethernet. The device driver software uses DMA (Direct Memory Access) to transfer the data to the Ethernet card for output over the bus and to read data from the card. While the GSFC Ethernet card has the capability to use traditional interrupt driven I/O for transferring the data, DMA was chosen for performance reasons. First, more bytes are sent in less time, and second, using DMA frees the CPU/Memory resources, which is in short supply on a spacecraft.

Network Services: Other common services provided by onboard applications include a reliable command link and reliable file delivery. Reliable commanding is typically implemented using the CCSDS Communications Operation Procedure-1 (COP-1). This Data Link Protocol has a retransmission control mechanism that provides a function of retransmitting lost or corrupted data to ensure delivery of data in sequence, without gaps or duplication over a Spacelink. While not a requirement of the onboard network, the first phase of the prototype activity includes many "COP-1-like" features including sequence verification, retransmit, bypass, and lock-out. It is anticipated that follow-on activities will address the flight component of COP-1 in the Data Link/Convergence layer software for the Spacelink.

In the baseline software, COP-1 functionality was handled by an application task called Command Ingest (CI). For the prototype, however, the bypass flag, control command flag, and sequence counter, normally defined in the CCSDS transfer frame, were added to the secondary header of the CCSDS command packet. The CLCW, used for verification, is maintained by CI in its original format. The CLCW is downlinked in a UDP packet where it is processed by the ground system, completing the reliable command link.

In addition to general commands executed in software, missions often have a need for "special" commands, which are executed by hardware and provide a minimal set of capability if there are problems with the spacecraft that cannot be resolved with general commanding. These commands are handled at the Physical and Data Link Layers. The prototype hardware supports special commands by assigning special UDP/IP destinations and ports to the packet. The controller on the NIC will look for this UDP port and process the command. Special commands are broadcast over the network.

Reliable File Transfer applications, such as CFDP and MDP, are easily implemented by interfacing the onboard component directly to the IP stack, as shown in Figure 7.

Figure 7. Prototype Configuration

Summary

At the writing of this paper, one thing had already become clear, the initial premise that the use of commercially available components for development and test is a valid one. While waiting for the spacecraft NIC and switch, it was easy and inexpensive to assemble a basic software system using commercial network cards and Linux device drivers. In addition, the use of the commercially available Ethernet Protocol engine has saved countless dollars in the development of the Flight Ethernet Network interface card design, both through the use of a standard MAC core and the ability to use existing standards and methods as templates for our flight applications. Future cost savings will also be realized by using commercial network cards and switches in system breadboards and ground test equipment.

While the prototype is not a complete, ready-for-flight solution, enough has been developed to mitigate the risks of using Ethernet as an onboard spacecraft bus. Future work includes supporting 100Mbps Flight physical layer, adding standard network support for MIL-STD-1553, and the development of a Quality of Service (QoS) implementation for flight Ethernet.

Acknowledgments

References

[1] Spacecraft Onboard Interfaces - Concept and Rationale, Consultative Committee for Space Data Systems, Green Book CCSDS 830.0-G-0.4, 2002